FDM Group

Trainer Skills & Availability

Group 1

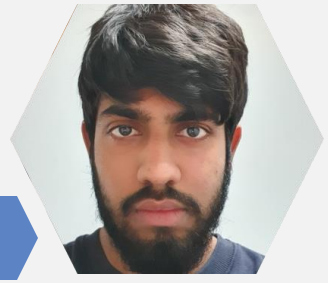
Requirements Elicitation

https://www.youtube.com/watch?v=8_lhUqEfBFc

Agenda

- Meet the team
- Use cases
- Functional & Non-Functional Requirements
- Risk Assessment
- Closing

Meet The Team

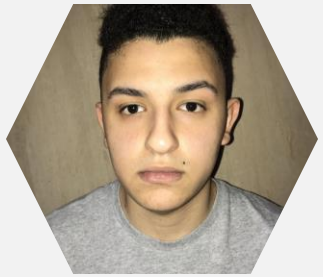


Hamzah

BSc Computer Science

Lucas

BSc Computer Science



Omar

BSc Computer Science with Management

Ibrahim

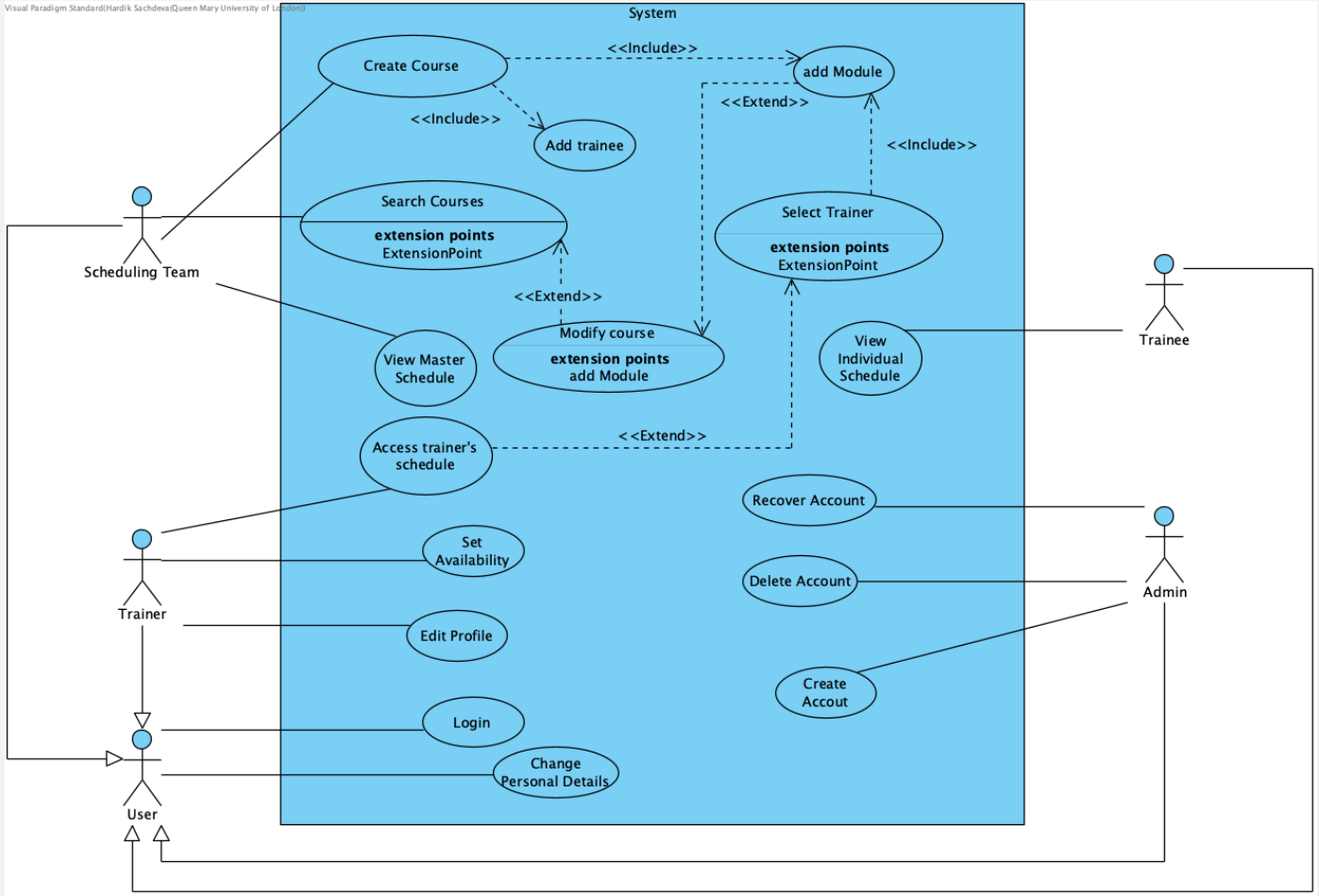
BSc Computer Science

Hardik

BEng Computer Systems Engineering

Use Case Diagram

- Current users: Trainee, Trainers, Scheduling Team, Admin
- All Users: can login, change personal details
- Trainee: Can view their schedule
- Trainers: Can set availability, view schedule, edit their profile
- Admin: Remove Account, Recover Account, Create a new Account
- Scheduling team: Can create new courses, add modules to the new and existing courses, view master schedule.



Changes After Domain Analysis

- Added a new user ”trainee” they can look at timetable, see who their trainer is and look at what room they have a booked course
- The Trainer can add their availability
- The scheduling team can add trainees to a course
- The scheduling team can select the room where the course will be booked in, and they can also select to have the course running online

Key Functional Requirements

Let’s dive in

ALL USERS

LOGIN

- System must ensure all users have a unique username
- System must allow all users to login to the system with a password

CREATE COURSE

- System should allow scheduling team members to be able to create courses
- System should ensure each course has a unique ID and a description
- INCLUDE ADD MODULE
- INCLUDE ADD TRAINEES

CREATE COURSE INCLUDES:

ADD MODULE

- Scheduling team members can add modules to the courses
- System should ensure each module has a unique ID and a description as well
- INCLUDE SELECT TRAINER
- System must ensure scheduling team member adds a room to the modules where they will be taught
- System must allow scheduling team members to search for room availability
- System must ensure each room has a unique ID and must store the location of each room
- Each module must have a unique ID and a description as well

ADD TRAINEE

- System must ensure scheduling team members add trainees to a course.
- System must show all modules for a course a trainee has been added to on that trainee's schedule

SELECT TRAINER

- INCLUDE ACCESS TRAINERS SCHEDULE

Access Trainer Schedule

System must allow scheduling team members to access and view trainers' schedules

- System must allow scheduling team members to add trainers to each module. The trainer must teach this module

- When a trainer is added to a module, the system must show that module on the trainer’s schedule

TRAINER FUNCTIONALITIES

SET AVAILABILITY

- System must allow trainers to add available hours to their own trainer schedule

ACCESS TRAINER SCHEDULE

- System must allow trainers to view their schedule
- System must allow trainers to check the room in which a module is booked

TRAINEE/CONSULTANT

VIEW INDIVIDUAL SCHEDULE

- System must allow trainees to view which trainer teaches which module

- System must allow trainees to check the rooms a module is booked in

ADMINISTRATOR

CREATE ACCOUNT

- System must allow admin to create new user accounts

DATA REQUIREMENTS FOR SYSTEM USERS

- System must store first name and last name for all trainers and trainees
- System must ensure all users have a unique username
- All users must be able to log in to the system with a password
- System must allow trainers to view and print their schedule
- System must allow trainees to view and print their schedule

- System should ensure each course has a unique ID and a description
- System must ensure scheduling team members add modules to each course
- System must ensure scheduling team members add trainees to a course

- System should ensure each module has a unique ID and a description as well.
- System must allow scheduling team members to add trainers to each module The trainer must teach this module
- System must allow trainees can view which trainer teaches which module
- System must ensure scheduling team member adds a room to the modules where they will be taught
- System should allow trainees to check the rooms a module is booked in
- System must allow trainers can check the rooms a module is booked in
- System must ensure each room has a unique ID and must store the location of each room

Non-Functional Requirements

Accessibility

- Web application should be adaptable to mobile browsers
- System must be capable of running on the latest versions of Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge
- High contrast between text and background

Capacity

- Database should be capable of storing up to 20,000 users
- System must be able to process 10,000 concurrent users

Performance/Reliability

- Schedule page's response time should not exceed 3 seconds
- System must not allow the creation of a course with no modules
- System must not allow room bookings for a module with no trainees

Security

- System should allow only scheduling team members and trainers to view a trainer schedule
- Web application must meet GDPR compliance rules, especially regarding data privacy
- Password must be 8 characters long containing at least one upper case letter, one lower case letter and a special character.

Risk Assessment

Risk	Likelihood	Severity	Impact	Preventative / Mitigating Actions
Poor time management	High	High	Unfinished product.	Evaluate progress frequently and adjust project plan if necessary
Missing member(s) due to health issues	High	Medium	Unfinished/ low quality product.	Have frequent meetings and communication to make sure all members aware of each other's progress.
Project team lack of understanding of the requirements	Medium	High	Product doesn't support requirements.	Have very accurate documentation for each step of the development.

Risk	Likelihood	Severity	Impact	Preventative / Mitigating Actions
Client contact leaves the company	Low	Medium	Product is delayed.	Establish a second representative from our client's company which could replace the first.
Devices used by the development team aren't stable	Low	Medium	Product is delayed/ progression is lost.	Equip the development team with reliable devices and make sure to backup data regularly.
Client has erroneous expectations from the project	Low	High	Low quality product.	Have a very well-defined data gathering method in order to make sure all the required information given by the client have been recorded.
System is unable to be used in the organisation	Low	High	Unusable final product.	Communicate clearly with the stakeholders the intent and function of the application.

Risk	Likelihood	Severity	Impact	Preventative / Mitigating Actions
Technologies used in development are not compatible	Low	High	Incomplete product.	Research all technologies that will be used during development and ensure they are compatible and adjust if they are not.
Development environment problems	Medium	High	Product finish date is delayed.	Confirm that each member of the development team is comfortable with the environment and establish a second development environment as a backup.
Incorrect requirement priorities	High	Medium	Product is delayed.	Create a clear list of requirements with their respective priorities so that the development team is always focusing on the most important requirements first.

Conclusion

In conclusion, we identified our system requirements, created our use case diagram and identified the possible risks of our project.

This documentation will all be used in the next project development phases of our system.